

US-PAT-NO: 6370573

DOCUMENT-IDENTIFIER: US 6370573 B1

TITLE: System, method and article of manufacture for managing
an environment of a development architecture framework

----- KWIC -----

Abstract Text - ABTX (1):

A system, method and article of manufacture are provided for managing an environment in a development architecture framework. Service of a system is managed based on service level agreements and/or operations level agreements. A plurality of system management operations are performed. The system management operations include start-up and shut-down operations, back-up and restore operations, archiving operations, security operations, and performance monitoring operations. Service is planned in order to anticipate and implement changes in the system.

Brief Summary Text - BSTX (10):

A system, method and article of manufacture are provided for managing an environment in a development architecture framework. Service of a system is managed based on service level agreements and/or operations level agreements. A plurality of system management operations are performed. The system management operations include start-up and shut-down operations, back-up and restore operations, archiving operations, security operations, and performance monitoring operations. Service is planned in order to anticipate and implement changes in the system.

Brief Summary Text - BSTX (12):

In one aspect of the present invention, the archiving operations may be performed to transfer data between different mediums with different compression ratios. In another aspect of the present invention, the performance monitoring operations may be performed to determine if resources of the system are sufficient to meet a desired performance level.

Detailed Description Text - DETX (40):

To date, Web development tools have been limited in their ability to create dynamic Web applications which span from client to server and interoperate with existing computing resources. Until recently, HTML has been the dominant technology used in development of Web-based solutions. However, HTML has proven to be inadequate in the following areas:

Detailed Description Text - DETX (74):

Relying on the Business Integration Methodology and its project organization

guidelines (0940--Organize Project Resource Task Package), the following should be prepared:

Detailed Description Text - DETX (79):

Accountability--How and by whom the performance will be measured

Detailed Description Text - DETX (99):

The Information Management team is responsible for ensuring that the project's knowledge capital and information resources are managed effectively. This includes:

Detailed Description Text - DETX (103):

Information Management encompasses Repository management, but generally has a broader scope than merely the repository contents, because most repositories are not capable of holding all the information resources of a project. It is, for example, common to have key project information reside in a combination of repositories, teamware databases, flat files, and paper documents. It is the Information Management team's responsibility to ensure consistency across all these formats. The responsibilities of the Information Management team therefore cover:

Detailed Description Text - DETX (109):

In addition to managing the information for the System Building team, the Information Management team must also manage the information resources of the other management processes--quality management, environment management, and project management.

Detailed Description Text - DETX (136):

Program Performance Reporting

Detailed Description Text - DETX (137):

Resource Management

Detailed Description Text - DETX (143):

Resource consumption

Detailed Description Text - DETX (153):

To optimize processes

Detailed Description Text - DETX (158):

Measuring and monitoring progress using established processes to ensure that a capability release is delivered on time, within budget, and that it meets or exceeds expectations.

Detailed Description Text - DETX (160):

Ensuring that resources are used effectively across projects for the release.

Detailed Description Text - DETX (245):

A vast amount of information is generated within the development environment, which needs to be carefully managed (for example, design documentation, application code, media content, test plans and test data). Information Management generally involves Repository Management, Folder Management and, where applicable, Object Management and Media Content Management. Since a number of teams rely on the service provided by the information management team, it is important that the level of service to be provided be chosen carefully, documented, and communicated. The arrangement should take the form of a **Service Level Agreement (SLA)**. Such an **SLA** typically defines how quickly a new data element is created and how repository changes are communicated. More generally it defines the division of responsibilities between the information management team and the other project teams at a detailed level.

Detailed Description Text - DETX (256):

As many repositories do not provide sufficient versioning functionality, it is common to have more than one repository on large projects. Typically, there may be one repository for development, one for system test, and one for production. This allows better control, but also requires significant resources to move repository objects from the development environment to the system test environment. By merging the development and system test repositories, the medium-sized project has a potential for productivity gains. If these gains are to be realized, great care must be taken when making corrections during system test. As a common repository is shared, any error analysis involving repository objects must take into account the possibility that these objects could have changed since the previous migration to system test. This situation can be managed by meticulously maintaining a comprehensive change log.

Detailed Description Text - DETX (265):

Analysis, reporting, and querying

Detailed Description Text - DETX (272):

Whenever a data element is changed, impact analysis must be performed to understand the side-effects. Where-used reports are useful to determine these side-effects. The repository manager must be able to obtain the list of direct references and the list of all components affected indirectly (transitive closure). In the latter case, a message based on a record containing a group, which makes reference to a changed data element is considered to be indirectly affected by the change. When adding a data element, no functional equivalent must exist, because redundancy creates difficulties for impact analysis and future maintenance.

Detailed Description Text - DETX (274):

The objects related to dialog definitions, reports, messages, and so forth, are usually maintained by the designers and programmers. When the dialogs and report programs are tested, approved, and ready to be promoted to the system test environment, the related objects must be locked. This is the responsibility of the Repository Management team.

Detailed Description Text - DETX (277):

Detailed, project-specific standards should exist for defining repository objects. These standards can form the basis for a repository validation program, which can run through the entire repository and report on detected deviations from standards. In some cases, this program can also enforce the standard.

Detailed Description Text - DETX (278):

Mass changes to the repository can be performed when the validation reports show the occurrence of many standards violations that follow a common pattern. This may occur in cases where:

Detailed Description Text - DETX (283):

Analysis, Reports, and Queries

Detailed Description Text - DETX (284):

Certain reports should be run daily, such as the list of new data elements or modified data elements. These reports can serve as an audit trail of changes and can be used to communicate changes to the entire team. Procedures should specify which reports are run daily and what their distribution should be.

Detailed Description Text - DETX (286):

When supporting specific kinds of repository analysis, the Repository Management team can provide custom reports or ad hoc queries that satisfy particular needs.

Detailed Description Text - DETX (358):

To fine-tune the development process, the important quality attributes must be measured. Sample metrics include:

Detailed Description Text - DETX (386):

Identify the resource allocation process

Detailed Description Text - DETX (393):

Allocate resources and scheduling

Detailed Description Text - DETX (394):

Monitor effort

Detailed Description Text - DETX (396):

While maintaining quality at a program level, the Quality Management team must liaise with each of the organizational units within the development environment in order to monitor the quality management processes within these units.

Detailed Description Text - DETX (400):

For more details, refer to Consistently Delivering Value: The CMM--How to Help Your Project Measure Up.

Detailed Description Text - DETX (416):

0710--Obtain and Deploy Resources

Detailed Description Text - DETX (432):

Project Management focuses on providing specific deliverables through balanced management of scope, quality, effort, risk, and schedule. Project Management processes follow a cycle of planning the project's execution, organizing its resources, and controlling its work. The Project Management team oversees all other teams within the development environment.

Detailed Description Text - DETX (436):

0940--Organize Project Resources

Detailed Description Text - DETX (509):

Management of Service Level Agreements (SLAs)

Detailed Description Text - DETX (512):

Service Level Agreements

Detailed Description Text - DETX (513):

In order to plan and organize the development work appropriately, a Service Level Agreement (SLA) must be in place between the Service Management group (typically part of the Environment Management team) and the developers. As with all other components of the development environment, this agreement should be kept simple. It should specify the following:

Detailed Description Text - DETX (518):

Specifications of service levels should be precise and the service must be measurable. The SLA should also specify how to measure this service (for example, system response times, request service times, backup frequencies). In

addition, the SLA must be managed. It may have to be modified as the environment changes, and it must be reviewed with developers on a regular basis to see if the service level is adequate.

Detailed Description Text - DETX (551):

Defining the SLA, with its specific, measurable criteria, is the basis for continuous improvement. The continuous improvement effort may focus on providing the same level of service with fewer resources, or on providing better service. An important part of quality management is ensuring that the Environment Management team understands the key performance indicators for service delivery, that these indicators are monitored, and that all personnel are adequately equipped with the tools and training to fill their responsibilities. While the entire team is responsible for delivering quality, the responsibility for Quality management should be assigned to a specific individual on the Environment Management team.

Detailed Description Text - DETX (569):

The Environment Management team must systematically monitor the development environment to ensure that it is stable, provides adequate response times, and satisfies the needs of the developers. This monitoring involves looking at trends and extrapolating them to anticipate problems with disk capacity, system performance, network traffic, and so forth.

Detailed Description Text - DETX (582):

In the development environment, it may be possible to outsource certain Systems Management tasks. For example, the LAN supplier may be willing to take responsibility for LAN support, upgrades, and so on. Similarly, an existing data processing center may be willing to take responsibility for host operations. Such agreements are very beneficial and make it possible to use project team members more effectively. However, outsourcing the development environment carries a risk, which can be mitigated by defining a Service Level Agreement with the provider. This will generally be very similar to the SLA established between the Environment Management team and the developers. One important difference is that punitive measures (to be applied if the SLA is not respected) must be specified to ensure that outside suppliers are strongly motivated to abide by the agreement.

Detailed Description Text - DETX (591):

Once the SLA is defined, the resources required for delivering the service can be specified. Questions to address include the staffing of these resources and training to ensure that they are equipped to deliver service as agreed.

Detailed Description Text - DETX (657):

Window and report design standards

Detailed Description Text - DETX (768):

Benefits Realization Test--The benefits realization test tests that the

business case for the system will be met. The emphasis here is on measuring the benefits of the new system, for example: increased productivity, decreased lead times, or lower error rates. If the business case is not testable, the benefits realization test becomes more of a buyer signoff.

Detailed Description Text - DETX (769):

Ideally, benefits realization test occurs prior to complete deployment of the system and utilizes the same environment that was used for the service-level test piece of operational readiness test. Tools are put in place to collect data to prove the business case (e.g., count customer calls). A team of people to monitor the reports from the tools and prove that the business case is achieved is still needed. The size of the team depends upon the number of users and the degree to which tools can collect and report the data. The benefits realization test tests that the business case for the system will be met. The emphasis here is on measuring the benefits of the new system, for example: increased productivity, decreased lead times, or lower error rates. If the business case is not testable, the benefits realization test becomes more of a buyer signoff.

Detailed Description Text - DETX (779):

Service level test--ensures that once the application is rolled out, it provides the level of service to the users as specified in the Service Level Agreement (SLA).

Detailed Description Text - DETX (782):

The operational readiness test is the point in the development process where all the application development, architecture development, and preparation tasks come together. The operational readiness test ensures that the application and architecture can be installed and operated in order to meet the SLA.

Detailed Description Text - DETX (838):

Resource reservation (for example, meeting rooms)

Detailed Description Text - DETX (840):

Status reports/meeting minutes

Detailed Description Text - DETX (858):

Managing resource usage (for example, disk space)

Detailed Description Text - DETX (863):

Group scheduling tools help to centrally manage the personal schedules of a group of people. This offers the advantage of being able to coordinate events that require the participation of a number of people automatically by checking 'group availability' rather than checking with each person individually. These tools may also be used to schedule other resources such as meeting rooms and

equipment. For the use of group scheduling tools to be successful, the personal schedules of each member of the group must always be current. This is the responsibility not only of the group scheduler, but also of the individuals involved.

Detailed Description Text - DETX (921):

Reporting

Detailed Description Text - DETX (931):

Reporting

Detailed Description Text - DETX (940):

Mobile code security--protects corporate **resources**, computer files, confidential information, and corporate assets from possible mobile code attack.

Detailed Description Text - DETX (988):

h) Does the tool provide repository **reporting** facilities?

Detailed Description Text - DETX (989):

Repository **reports** serve as an audit trail for changes to objects within a repository and can be used to communicate these changes to the entire team. The Repository Management tool should provide this utility.

Detailed Description Text - DETX (990):

Reports for impact analysis are extremely useful in the change control process. As the repository maintains relationships between repository objects, 'where-used' and 'contains' **report** facilities can be very useful when dealing with change requests.

Detailed Description Text - DETX (1045):

m) Analysis, **Reports**, and Queries

Detailed Description Text - DETX (1046):

Reports for impact analysis are extremely useful in the change control process. As the repository maintains relationships between repository objects, where-used and contains **reports** are usually provided with the repository. Storing the names of affected repository objects in an area-affected table can be useful when grouping change requests during assignment, or when defining a release. The area-affected table is also a valuable tool that can be used to facilitate migration from development to system test.

Detailed Description Text - DETX (1047):

The ability to easily create various repository **reports** is important to

leverage the information in the repository. A scripting language, a simple report builder, or a query tool provides this capability. Having a query tool with an intuitive user interface and good report formatting features is a necessity on a large project. The query tool can be used to provide standard reports for designers and programmers, printed design information for external reviews, and ad hoc requests for the repository administrator.

Detailed Description Text - DETX (1081):

Metrics are an important part of quality management in that they provide a method of measuring (for example, sampling, testing, and determining) whether a process or product meets a given criterion. With Metrics, different stakeholders can agree that a product objectively meets an expectation, or that a process has been improved by a measurable amount. Without Metrics, stakeholders can only have subjective opinions that may or may not agree.

Detailed Description Text - DETX (1082):

Measurement tools are used to measure process quality and product quality. Process quality may include Metrics such as the time it takes to process a change request. Product quality should be measured for all the product expectations the project has set. This measurement process is the inspection part of quality management.

Detailed Description Text - DETX (1093):

Program and Project Management tools assist the management teams in their daily work. These tools, typically packaged as integrated suites of software, provide the basic functionality required for planning, scheduling, tracking, and reporting at both the program and project level.

Detailed Description Text - DETX (1095):

Planning tools are used to assist in program and project planning including the development of the Program Resource Plan, the Work Breakdown Structure (WBS), the Organization Breakdown Structure, Cost Accounting, milestones, and deliverables.

Detailed Description Text - DETX (1097):

Scheduling Tools are used to allocate resources against the WBS, to determine the timeline for a specific project, and to schedule the allocation of resources at the program level.

Detailed Description Text - DETX (1099):

Project tracking tools enable the project manager to track the actual project status against the original plan and schedule. Integration with the time reporting system and techniques such as Estimates to Complete (ETCs) are valuable in tracking project status.

Detailed Description Text - DETX (1100):

Reporting

Detailed Description Text - DETX (1101):

Reporting Tools are used to summarize status and metrics to program and project management.

Detailed Description Text - DETX (1125):

An engagement team must determine whether to purchase a Configuration Management tool or build one. The build decision should consider the cost of designing and developing the functions required by the engagement team. Additionally, the project must consider the **resources** and development time required to build the tool and when the tool is needed in the application development schedule.

Detailed Description Text - DETX (1136):

e) Does the tool provide capabilities for exception **reports**?

Detailed Description Text - DETX (1137):

If for some reason a repository component is not at the correct promotion level, the tool should be able to **report** on this when required.

Detailed Description Text - DETX (1144):

The tools should automate the storage and retrieval of all dependent software components indicated by an impact analysis **report**.

Detailed Description Text - DETX (1150):

Version Control tools allow systematic storage of information about who makes changes in what order so that the evolution of the system can be tracked. The tools usually provide a facility to **report** on differences in versions so the version that existed when a critical change was made can be identified and recreated or retrieved. The tools can also provide a means of documenting why decisions are made during the evolution of the system. These decisions would have been made based on the version of the documentation for the system that existed at that time. Version Control tools allow the state of the system at a particular time to be recorded. Hence improved auditability for decisions can be achieved.

Detailed Description Text - DETX (1174):

Flexible, customizable sorting and **reporting** to ensure that a change is handled in a timely manner

Detailed Description Text - DETX (1197):

Change requests may occur as a consequence of changing requirements, or as a result of nonconformities (or defects) in the system. The tool should be able to classify change requests into categories such as incidents, faults, or

enhancements. The tool should also have the ability to update these categories if required. Classification of different change requests in several different ways such as area affected, priority, estimated cost or authorization is important to ensure correct scheduling of the implementation of changes. Flexible, customized sorting and **reporting** based on this classification is required to ensure that change is handled in a timely manner.

Detailed Description Text - DETX (1200):

If an impact analysis tool cannot be found that supports the entire environment, it is critical to develop procedures or utilities that will **report** on where items are used. The first step is to identify the items to be searched, and to build procedures around searching them (for example, databases, files, workspaces, programs, screens/forms, **reports**). It is also important to identify who will be responsible for the impact analysis (DBA, analysts, programmers, team leaders, and so on) to avoid this work falling between the cracks.

Detailed Description Text - DETX (1204):

The tool should allocate change requests to different releases based on priority and **resource** availability. It should also provide a means of attaching a deadline to a change request.

Detailed Description Text - DETX (1208):

The tool should provide a capability to generate exception **reports** that highlight issues such as change requests that are in danger of not meeting the release to which it was allocated.

Detailed Description Text - DETX (1215):

- i) What **reporting** capabilities are needed on the project?

Detailed Description Text - DETX (1216):

Some Change Control tools can **report** on status of change requests at the individual, team, and project level. Such **reports** can provide information about work done to date and Estimate to Complete (ETC) values.

Detailed Description Text - DETX (1250):

Monitoring functionalities, in order to **measure** progress towards delivery goals

Detailed Description Text - DETX (1267):

Tools to support the creation, management, and **reporting of Service Level Agreements (SLAs)** and Operations Level Agreements (OLAs)

Detailed Description Text - DETX (1290):

Performance Monitoring tools help ensure that the available **resources** are

sufficient to meet the developers' performance requirements. These tools can be used to assess end-to-end performance of both batch processes such as backups, and interactive processes such as repository-based file retrieval.

Detailed Description Text - DETX (1307):

Problem Management tools log information about problems detected, classify them, and generate reports. This is essential for capturing metrics information.

Detailed Description Text - DETX (1324):

Stage containment is an approach to identify problems in the system before they pass to the next stage. It is a measure that helps build quality into the system. The goal of stage containment is to minimize the number of errors being passed to the next stage. For the purpose of stage containment, problems are sorted into categories. Errors are defined as problems found in the stage where they were created. Defects are problems found in a stage successive to the stage where they were created. Faults are problems found in production. The longer a defect remains undiscovered, the more difficult and expensive it will be to correct. Because each stage relies on the decisions made during the creation of the specification in the previous stage, detecting an error in a stage after it was made may invalidate some or all of the work done between the time the issue was created and the time it was discovered.

Detailed Description Text - DETX (1340):

Problem Management tools log error information, generate error reports (such as System Investigation Reports or SIRs), classify problems, and record information on the source of the error. Problem Management tools are essential for the capture of stage containment metric information.

Detailed Description Text - DETX (1392):

It is important to consider the various utilities available with the data modeling tools. Two such utilities include impact analysis and reporting. Impact analysis capabilities allow the user to understand the impact of a change to the data model. Impact analysis functionality is one of the key tools used by engagement teams to assist with change management and change control activities. Some products will also include report generators which are useful for generating data and attribute definition reports as well as ad hoc reports.

Detailed Description Text - DETX (1406):

The information management component may provide the security needed in a multi-designer environment. If this is not the case then a multi-designer data modeling tool should be used. The tool may provide a central dictionary which allows design data to be shared between several designers and includes security checks to monitor any conflicts in overlapping access rights between designers.

Detailed Description Text - DETX (1489):

Dedicated performance modeling tools should be considered for any project that involves high transaction volumes or a complex architecture with several platforms. Performance is critical for such systems and a performance model is required in order to predict and optimize that performance.

Detailed Description Text - DETX (1503):

Specific modeling tools can provide advantages such as cross referencing (for example, are all the methods used in the Interaction diagrams described in the class definitions?), automatic propagation of changes to other diagrams, generation of reports, and generation of skeleton code. However, some tools have problems with:

Detailed Description Text - DETX (1529):

Prototyping can address this problem by simulating key user interface components, thus enabling the development team to measure the usability of the proposed system at a very early stage. The most important quality of a prototyping tool is its development speed. If prototyping can be performed in hours or days rather than weeks or months, it becomes possible to perform more iterations, which explore different options. This may lead to a much better system, given that the user's perception matures with each iteration. This, in turn, improves the quality of user input.

Detailed Description Text - DETX (1552):

If the system is to be used by dedicated people where the measure of productivity is solely the number of transactions they can get through per second, then user interface prototyping tools are important. Prototyping tools provide a means of getting to the easiest and most efficient interface. Prototyping tools facilitate selection between alternative styles of interaction and provide a means of addressing performance issues.

Detailed Description Text - DETX (1649):

Presentation design tools provide a graphical depiction of the presentation layer of the application, such as windows, dialogs, pages, navigation and reports. Tools in this category include window editors, report editors, and dialog flow (navigation) editors. Window editors enable the developer to design the windows for the application using standard GUI components. Report editors enable the developer to design the report layout interactively, placing literals and application data on the layout without specifying implementation details such as page breaks. The majority of these tools generate the associated application code required to display these components in the target system.

Detailed Description Text - DETX (1681):

It is important to determine how well the product integrates with other design and development tools, presentation services (graphics, multi-media, etc.), data access services (databases and database API libraries), distribution services (distributed TP monitor), transmission services (SNA, HLLAPI, etc.), data dictionary, desktop applications, and programming languages

for call-out/call-in. Additional consideration should be given to add-on and third-party products/enhancements such as specialized widgets, report writers and case tools.

Detailed Description Text - DETX (1701):

Special characters, differences in field lengths, and differences in number formats are some of the things that contribute to the complexity of a multi-language application. Window and report design are among the areas affected by differences in the language used for presentation.

Detailed Description Text - DETX (1705):

The presentation design tools should be tightly integrated with the system components stored in the repository, such as windows, reports, screens, and other more abstract models to ensure consistency.

Detailed Description Text - DETX (1745):

Developers use interactive navigation tools to identify requirements for a new system from the functionality and design of a legacy system. These tools enable the developer to interactively and graphically navigate the legacy system, determining the system's characteristics such as system structure, module flow, flow control, calling patterns, complexity, and data and variable usage. An alternate form of presentation is through reports. These provide cross-reference listings or graphical representations of control or data flows.

Detailed Description Text - DETX (1749):

An extraction tool, in conjunction with a repository population tool, enables the developer to reuse selected portions of a legacy system. The extraction tool can typically read and extract information from source code, screens, reports, and the database. The most common information extracted from a legacy system, however, is the data: record/table structure, indexes, and data element definitions.

Detailed Description Text - DETX (1758):

Packaged components are generally third party components that provide ready-made business logic that is customizable and reusable. These can range from simple components offering limited functionality (for example, worksheet or charting GUI components), to components that handle a significant portion of the application architecture (for example, data access components and firewalls). The advantage of using such components is that they have already been coded, tested, optimized, and documented.

Detailed Description Text - DETX (1768):

Components are often developed with a preferred platform in mind. Components optimized for one platform may have severe performance problems on others. If performance is a factor (and it nearly always is) ensure that components are designed specifically for the platform of the target system.

Detailed Description Text - DETX (1777):

Construction tools are used to program or build the application: client and server source code, windows, reports, and database. Along with the onset of Visual Programming, the more traditional form of construction tools have been superseded by Integrated Development Environments (IDEs) which take all the basic components required for construction, and integrate them into a single system. Although IDEs are now the preferred tools for most construction, the components that make up these tools remain the same--Source Code Editor, Compiler/Linker/Interpreter, Generation Tools and Debugging Tools.

Detailed Description Text - DETX (1854):

Generators are used to leverage the powers of code reuse and code regeneration. The ability to reuse code reduces both the time and resources required on a project. Code regeneration eases maintenance issues by propagating changes throughout multiple sections of code.

Detailed Description Text - DETX (1857):

The code/applications generated by the tools vary in performance. Optimized code usually results in faster run times. It is important to identify the high priority components that will benefit most from the tool.

Detailed Description Text - DETX (1864):

Code Analysis--Code analysis provides the objective information and metrics needed to monitor and improve code quality and maintenance (e.g. static analyzer, documentor, auditor).

Detailed Description Text - DETX (1870):

Code and Object libraries provide the developer with ready-made components (such as GUI components or simple utilities), which may be integrated into architecture or application code. The advantage of using such components is that they have already been coded, tested, optimized, and documented.

Detailed Description Text - DETX (1911):

Year 2000 Testing Contacts and KX Resources

Detailed Description Text - DETX (1970):

Automating the execution of a non-repeatable test model is a waste of resources, as the test tool will not be able to re-execute the tests automatically or perform full regression tests with little effort. Little or no benefits will be achieved from automation.

Detailed Description Text - DETX (1979):

A reporting option is planned to produce metrics and management type reports.

Detailed Description Text - DETX (2039):

Performance Management tools support application performance testing. Owing to the large number of components in modern systems, performance modeling can be a complex task and requires tools to effectively manage the process. These tools monitor the real-time execution and performance of software. They help to maximize transactions and response time to the end user. They are also useful in identifying potential bottlenecks or processing anomalies.

Detailed Description Text - DETX (2040):

In the case of Internet-based applications, as the Internet is not a controlled environment, performance management tools can only measure performance within the domain of the controlled environment (up to the Internet Service Provider). However, in the case of intranet-based systems, where the environment is controlled from end-to-end, Performance Management may be performed across the entire system.

Detailed Description Text - DETX (2098):

Understanding the anticipated volumes will provide key input to sizing the system. Predicted business volumes stated in the SLA should be used to help determine the appropriate sizes for machines, databases, telecommunications lines, etc. Alternatively, experience from previous engagements can provide useful input.

Detailed Description Text - DETX (2106):

The presentation component provides the interface between the manager(s) of the system and management data generated by the system. Data can be manipulated for various forms of output. By integrating the operational architecture it is possible to reduce the number of front-end interfaces required. Commonly, the presentation component uses a GUI front-end interface. This component is also responsible for real-time and historical report generation.

Detailed Description Text - DETX (2110):

As with End User Services in the centralized model, the Help Desk is the single point of contact for all end users. This unit has end-to-end accountability for all user incidents and problems regardless of whether or not it has the resources to fix them (i.e., it must contact the necessary technical resources in either IS organizations to ensure the incidents and problems get resolved).

Detailed Description Text - DETX (2133):

As with End User Services in the centralized model, the Help Desk is the single point of contact for all end users. This unit has end-to-end accountability for all user incidents and problems regardless of whether or not it has the resources to fix them (i.e., it must contact the necessary technical resources in either IS organizations to ensure the incidents and problems get resolved).

Detailed Description Text - DETX (2143):

Incidents and requests should be closed with a date and time stamp to help trend analysis and service level reporting.

Detailed Description Text - DETX (2147):

If the Incident, Request and Problem management functions are to be centralized, these functions need to be able to control and monitor incidents and problems, but other functions should be able to gain access to input detailed technical information or progress updates. If Incident and Request management is distributed, it is recommended that remote locations are given access to the central system, rather than operating local systems. (Some problem areas are local sites operating on different time zones and standardizing escalation procedures from local sites.)

Detailed Description Text - DETX (2183):

In the event of a significant system failure, Disaster Recovery processes will be invoked to re-route the system resources to a secondary, stable configuration until the primary resources can be restored. Within a distributed environment, disaster recovery must account for differing levels of disaster whether at a central or distributed site(s).

Detailed Description Text - DETX (2186):

The way in which a disaster is defined will be dependent upon which resources are critical to the business. For example, a data center failure may be critical for one client whereas a server failure for another is more critical.

Detailed Description Text - DETX (2188):

This will be defined in detail within the SLA, but high level service recovery targets must be understood, so that high level recovery plans can, in turn, be produced.

Detailed Description Text - DETX (2195):

Hardware Maintenance maintains all of the components within a distributed system to protect the investment of the organization. Generally agreed upon in the SLAs, maintenance contracts are carried out, monitored and recorded for each asset as appropriate.

Detailed Description Text - DETX (2199):

Billing & Accounting also makes payments to service providers for services and equipment provided in accordance with agreed upon SLAs. As part of this payment process Billing & Accounting reconciles bills from service providers against monitored costs and SLA/OLA violations.

Detailed Description Text - DETX (2202):

Capacity Modeling & Planning ensures that adequate resources will be in place to meet the SLA requirements, keeping in mind operational requirements which may require additional capacity. Resources can include such things as physical facilities, computers, memory/disk space, communications lines and personnel. Through this component, changes to the existing environment will be determined, modeled and planned according to the necessary requirements.

Detailed Description Text - DETX (2221):

The tool should provide control dependencies to schedule workloads such as: Task/job sequence enforcement, external/internal event driven. Graphically displays work flow from the scheduling criteria and includes such information as task/job name, task description, average run time and resource requirements. Allow clients to define user schedules that can be based on predecessor events in the production environment. Reporting capabilities for forecasting, simulation and analyzing scheduled workload. Monitoring capability of past, present and future workloads as well as tracking of current workload termination notification of normal or abnormal completion.

Detailed Description Text - DETX (2227):

Communication with Performance management component to forecast resource requirements, such as near line storage, DASD space, and etc.

Detailed Description Text - DETX (2236):

It makes it easy to develop and maintain report

Detailed Description Text - DETX (2238):

Reports arrive to the addressee more quickly

Detailed Description Text - DETX (2239):

It is possible to sign reports electronically

Detailed Description Text - DETX (2247):

If spooling is available, printing can be handled as a background task, freeing up system resources for use on-line.

Detailed Description Text - DETX (2251):

Large print jobs may utilize system resources considerably (e.g., WAN, LAN, printer), and may tie up the printing queue for other individuals. This type of printing should be performed in off-hours or delayed to avoid contention for the printer during business hours.

Detailed Description Text - DETX (2256):

Controls report production and distribution from the moment the report is created to the time the printed report is dropped in the end-user's mailbox

(electronic, paper, microfiche, etc.)

Detailed Description Text - DETX (2260):

Provides for the archival of reports in a compressed format first on disk, for a user specified time and then to tape or optical.

Detailed Description Text - DETX (2261):

Process reports in due-out-sequence.

Detailed Description Text - DETX (2262):

Automatic report balancing and archives the balancing reports for easy auditor review.

Detailed Description Text - DETX (2264):

Provide report reprint capability, avoid reruns in lost report situations.

Detailed Description Text - DETX (2265):

Provide centralized management of report setup and delivery information

Detailed Description Text - DETX (2270):

Communicates with the recovery management facility to delete reports that will be recreated.

Detailed Description Text - DETX (2271):

Communicates report volumes to the resource consumption management facility.

Detailed Description Text - DETX (2273):

Support multiple printer types as well as report delivery across them. This includes printer format translation (PCL, Postscript, etc.) and code translation.

Detailed Description Text - DETX (2279):

System Startup and Shutdown performs the activities required for the startup or shutdown of the entire system (e.g., hardware, applications), or portions of the system depending upon the identified requirements. Within a distributed environment, the system includes both centralized and remote resources.

Detailed Description Text - DETX (2289):

Analysis of the system and other resources need to be addressed?

Detailed Description Text - DETX (2290):

The state of an application, the system or a specific resource must be known

at all times. Common activities performed as part of Startup/Shutdown include:

Detailed Description Text - DETX (2300):

Mass Storage Management involves those activities related to the handling of various types of centralized and distributed storage media including the monitoring and controlling of storage resources and their usage.

Detailed Description Text - DETX (2328):

Interface with the Capacity/Resource manager to create a definable resource forecast.

Detailed Description Text - DETX (2351):

Designed along the lines requester-server model; more specifically the tool runs on the server machine and acts as a shared resource for data access, integrity, security recovery, etc.

Detailed Description Text - DETX (2367):

Does the tool add color to MODE architecture model through performance measures?

Detailed Description Text - DETX (2401):

The software and data distribution mechanism itself updates either the software, data, or configuration information on a machine(s), reports the relative success/failure of the distribution and updates the asset information for the sites/machine(s) affected by the distribution.

Detailed Description Text - DETX (2406):

The existing skills must be assessed and a forward-thinking training direction must be defined. The training plan will likely emphasize newer technologies and different methods of training with the underlying goal of providing the appropriate level of service as required by the SLAs.

Detailed Description Text - DETX (2441):

Due to the number of components, users may be required to have multiple ID(s) and passwords unless the system is designed to allow a user to access all of the required resources through a single logon. As most products on the market typically allow access to only a subset of resources, single logons with multiple ID and password coordination may be difficult to achieve. Issues such as periodic required password changes can be difficult to overcome while maintaining adequate security.

Detailed Description Text - DETX (2444):

Protects all computer resources, facilities and data from accidental or intentional destruction, modification, disclosure and/or misuse.

Detailed Description Text - DETX (2447):

Maintains a security log and user profile of what was accessed when, from a computer resource, facility and data view point.

Detailed Description Text - DETX (2448):

Security Administration ability to monitor the activity of a user of resource.

Detailed Description Text - DETX (2490):

In addition to the production system data as describes above, it contains any existing release and release components such as software modules, documents and procedures. It also contains service level agreements and actual figures for user groups and devices, incidents, problems and change requests. It may also contain additional data such as performance data or log of all backups taken.

Detailed Description Text - DETX (2527):

Rollout Planning handles the greatest period of change in distributed systems management—system rollout and installation. During rollout every site and every user may be impacted by the changes taking place. Since delivery of the system will affect how well it is received by the users and is oftentimes defined by an SLA(s), delivery of the system must take place smoothly with minimal interruption to the users. This can be challenging when both old and new architecture domains must exist concurrently until the rollout has been completed.

Detailed Description Text - DETX (2551):

Confirmation of release scheduling and determine if the release is on schedule and report on progress of release.

Detailed Description Text - DETX (2569):

In addition to the production system data as describes above, it contains any existing release and release components such as software modules, documents and procedures. It also contains service level agreements and actual figures for user groups and devices, incidents, problems and change requests. It may also contain additional data such as performance data or log of all backups taken.

Detailed Description Text - DETX (2604):

Capacity Modeling & Planning ensures that adequate resources will be in place to meet the SLA requirements, keeping in mind operational requirements which may require additional capacity. Resources can include such things as physical facilities, computers, memory/disk space, communications lines and personnel. Through this component, changes to the existing environment will be determined, modeled and planned according to the necessary requirements.

Detailed Description Text - DETX (2607):

Capacity Planning & Modeling must coordinate the requirements across the system (e.g., networks, servers, workstations, CPU, etc.) Capacity is driven by the need to meet SLAs with the user communities and as part of the planning and modeling process, future threats to capacity should be identified.

Detailed Description Text - DETX (2612):

Performance Management ensures that the required resources are available at all times throughout the distributed system to meet the agreed upon SLAs. This includes monitoring and management of end-to-end performance based on utilization, capacity, and overall performance statistics. If necessary, Performance Management can make adjustments to the production environment to either enhance performance or rectify degraded performance.

Detailed Description Text - DETX (2618):

Performance Management needs to consider performance from a business perspective, not merely a systems one. Most transactions in distributed systems utilize a wide variety of resources, and the measurement of end-to-end response time becomes the sum of the time expended by each one of the components sequentially involved in the transaction less the time while components were processing in parallel.

Detailed Description Text - DETX (2621):

Will performance be measured from end-to-end or merely for individual components?

Detailed Description Text - DETX (2625):

Will only selected transactions be measured, and if so, should this selection be configurable?

Detailed Description Text - DETX (2626):

It may be necessary to measure business critical transactions only; specified within the SLA. If the facility to select specific transactions is required, significant customization of the system maybe necessary.

Detailed Description Text - DETX (2630):

As SLAs will likely be tied in some way to performance, it is important to monitor and correct the systems performance as it degrades to ensure that operational levels are maintained and that the SLA(s) will not be violated.

Detailed Description Text - DETX (2633):

Collect, analyze and display in graphical format real-time performance characteristics from a wide range of resources. Analyze current workload and configuration data and forecast future requirements, as well as providing input into the Financial planning process.

Detailed Description Text - DETX (2636):

Monitoring capabilities include the ability to measure CPU and disk utilization, memory occupancy, transaction response time, reports (storage & distribution), printers, network utilization and performance, circuit utilization, backup facilities, WAN/LAN utilization..

Detailed Description Text - DETX (2639):

May require use of some or all of the following monitoring tools: operating system monitor, on-line monitor, batch monitor, data base monitor, (host, server) and network monitor (WAN, LAN).

Detailed Description Text - DETX (2641):

Performance measures must be consistent with Service Level management techniques

Detailed Description Text - DETX (2643):

Resource utilization statistics may be used to generate costing, and potential billings for customers.

Detailed Description Text - DETX (2644):

Passes data to the resource consumption management facility to report on the recurring processing cost of each business application.

Detailed Description Text - DETX (2665):

To ensure that the Operability Principles have been satisfied, each release should, in principle, undergo a release test of a full business cycle (to show that Operations can run it) and full business volumes (to show that SLA targets can be achieved). These tests are, however, expensive in terms of dedicated hardware requirements, people, and elapsed time.

Detailed Description Text - DETX (2691):

Store and forward techniques can help reduce the contention for system resources during business hours. Store and forward can also reduce the amount of traffic in the system based upon the routing tables defined within the system. Instead of having one machine send the same file to multiple machines, for instance, a cascading forwarding mechanism can be used. This also improves the system performance as files are sent a minimal number of times to certain devices which then forward the files on to other devices.

Detailed Description Text - DETX (2699):

In the event of a significant system failure, Disaster Recovery processes will be invoked to re-route the system resources to a secondary, stable configuration until the primary resources can be restored. Within a distributed environment, disaster recovery must account for differing levels of

disaster whether at a central or distributed site(s).

Detailed Description Text - DETX (2708):

Hardware Maintenance maintains all of the components within a distributed system to protect the investment of the organization. Generally agreed upon in the SLAs, maintenance contracts are carried out, monitored and recorded for each asset as appropriate.

Detailed Description Text - DETX (2730):

Performance Management ensures that the required resources are available at all times throughout the distributed system to meet the agreed upon SLAs. This includes monitoring and management of end-to-end performance based on utilization, capacity, and overall performance statistics. If necessary, Performance Management can make adjustments to the production environment to either enhance performance or rectify degraded performance.

Detailed Description Text - DETX (2758):

Will Equipment be resourced from multiple or single suppliers?

Claims Text - CLTX (2):

(a) managing service to a developer of the development environment based on at least one of service level agreements with the developer and operations level agreements with the developer;

Claims Text - CLTX (8):

5. A method as recited in claim 1, wherein the performance monitoring operations are performed to determine if resources of the system are sufficient to meet a desired performance level.

Claims Text - CLTX (10):

(a) a code segment that manages service to a developer of the development environment based on at least one of service level agreements with the developer and operations level agreements with the developer;

Claims Text - CLTX (16):

10. A computer program as recited in claim 6, wherein the performance monitoring operations are performed to determine if resources of the system are sufficient to meet a desired performance level.

Claims Text - CLTX (18):

(a) logic that manages service to a developer of the development environment based on at least one of service level agreements with the developer and operations level agreements with the developer;

Claims Text - CLTX (24):

15. A system as recited in claim 11, wherein the performance monitoring operations are performed to determine if resources of the system are sufficient to meet a desired performance level.